

UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CIÊNCIA DA COMPUTAÇÃO – Campus Chapecó
Disciplina de Construção de Compiladores

Descrição do PROJETO 2: Construção de Analisador Sintático

Braulio Mello
Última atualização: 02/05/24

Projeto: Reconhecedor Sintático

Objetivo:

Construção de um analisador sintático para reconhecimento de uma linguagem hipotética cujos tokens foram definidos no Projeto 1.

Etapa 1 (70% da nota):

- construir as regras sintáticas da linguagem: GLC;
- aplicar eliminação de inúteis e fatoração na GLC;
- Construção do conjunto de itens válidos, transições e follow
- Construção da tabela de parsing SLR ou LALR
 - SLR Parser Generator (<http://jsmachines.sourceforge.net/machines/slr.html>)
 - GoldParser (<http://www.goldparser.org/>)
- Implementação do algoritmo de mapeamento da tabela para reconhecimento sintático
- Gerenciamento da tabela de símbolos (suporte para as próximas etapas)
- Integrar procedimentos (ações semânticas – TDS) para os seguintes itens:
 - tratamento de erros (pelos menos o tipo de erro e linha de ocorrência);
 - adição de atributos aos símbolos (id de tokens reconhecidos);
 - valores nos atributos;
- entrada: fita de saída do reconhecedor léxico;
- saída: aceite ou mensagem(ns) de erro(s) sintático(s);
- adicionar informações na tabela de símbolos para uso nas etapas subsequentes (a definir em cada projeto).

Etapa 2 (10% da nota):

- definir uma característica semântica cujas informações adicionadas na TS sejam suficientes para a análise;
- implementar a análise semântica da característica especificada;
- entrada: código / tabela de símbolos
- saída: aceite ou mensagem(ns) de erro(s) semântico(s);

Etapa 3 (10% da nota):

- cada redução de uma produção (reconhecimento de uma estrutura sintática) é seguida de uma chamada a uma ação que gera o correspondente trecho de código intermediário com os respectivos rótulos dos identificadores, constantes, etc.
- basta aplicar a geração de código intermediário para uma das regras sintáticas (demonstração);
- entrada: reconhecimento de uma estrutura sintática conforme regras de produção.
- saída: código intermediário;

Etapa 4 (10% da nota):

- aplicar uma estratégia de otimização de código sobre o código intermediário gerado na etapa anterior
- entrada: código intermediário
- saída: código intermediário otimizado

Texto:

Elaborar texto entre 3 e 6 páginas contendo:

- Título, autores e instituição
- Resumo: breve apresentação do teor do texto
- Introdução: Contextualização sobre reconhecedores sintáticos (propósitos, aplicação e características), apresentação de problema e objetivo do trabalho.
- Referencial teórico: Breve explanação sobre os conceitos, técnicas e/ou teoremas fundamentais para o desenvolvimento do trabalho.
- Implementação e resultados: Apresentação dos detalhes de especificação/implementação do analisador sintático e estudo de caso para validação.
- Conclusões: O que foi feito, dificuldades, resultados finais e perspectivas para continuidade do trabalho (por exemplo, sugerir alterações futuras para utilizar a implementação no ensino de compiladores).

Apresentação:

- Trabalho individual ou em dupla
- A apresentação será na modalidade pergunta (docente) e resposta (discente). Não é necessário preparar apresentação (slides, etc).
- A nota será de acordo com o domínio dos detalhes do projeto prático demonstrado através arguição.
- A não apresentação do projeto resultará em nota 0 (zero), independente da entrega da implementação no SIGAA.
- Data de apresentação do projeto: **20/06/24**
- Data limite para apresentação do projeto: **04/07/24**
- Após **04/07/24** não serão mais aceitas apresentações do projeto.